

C++ ASSIGNMENT

1.Ques :Given an array of integers, sort it in descending order using merge sort algorithm.

Ans: `#include<iostream>`

`#include<vector>`

`#include<algorithm>`

`using namespace std;`

`void merge(vector<int> &a,vector<int>& b,vector<int>& res){`

`int i=0,j=0,k=0;`

`while(i<a.size()&& j<b.size()){`

`if(a[i]>b[j]){`

`res[k++]=a[i++];`

`}`

`else res[k++]=b[j++];`

`}`

`if(i==a.size()) while(j<b.size()) res[k++]=b[j++];`

`else while(i<a.size()) res[k++]=a[i++];`

`}`

`void mergesort(vector<int>& v){`

`int n=v.size();`

`if(n==1) return;`

`int n1=n/2,n2=n-n/2;`

`vector<int> a(n1),b(n2);`

`for(int i=0;i<n1;i++){`

`a[i]=v[i];`

`}`

`for(int i=0;i<n2;i++){`

`b[i]=v[i+n1];`

```

    }
    mergesort(a);
    mergesort(b);
    merge(a,b,v);
}
int main(){
    vector<int> arr={5,1,4,3,2};
    mergesort(arr);
    for(int i=0;i<arr.size();i++){
        cout<<arr[i]<<" ";
    }
}

```

2.Ques :Reverse Pairs (Leetcode Problem) : Given an integer array nums, return the number of reverse pairs in the array.

A reverse pair is a pair (i, j) where:

$0 \leq i < j < \text{nums.length}$ and
 $\text{nums}[i] > 2 * \text{nums}[j]$.

Ans: #include<iostream>

#include<vector>

#include<algorithm>

using namespace std;

int inversion(vector<int> &a,vector<int> &b){

int count=0;

int i=0,j=0;

while(i<a.size() && j<b.size()){

if(a[i]>2LL*b[j]){

count+=a.size()-i;

j++;

}

else i++;

```

    }
    return count;
}

void merge(vector<int> &a,vector<int> &b,vector<int> &res){
    int i=0,j=0,k=0;
    while(i<a.size() && j<b.size()){
        if(a[i]<b[j]) res[k++]=a[i++];
        else res[k++]=b[j++];
    }
    if(j==b.size()) while(i!=a.size()) res[k++]=a[i++];
    if(i==a.size()) while(j!=b.size()) res[k++]=b[j++];
}

void mergesort(vector<int> &v,int &c){
    int n=v.size();
    if(n==1) return;
    int n1=n/2,n2=n-n/2;
    vector<int> a(n1),b(n2);
    for(int i=0;i<n1;i++){
        a[i]=v[i];
    }
    for(int i=0;i<n2;i++){
        b[i]=v[i+n1];
    }
    mergesort(a,c);
    mergesort(b,c);
    c+=inversion(a,b);
    merge(a,b,v);
    a.clear();
    b.clear();
}

```

```
int main(){
    vector<int> arr={2,4,5,3,1};
    for(int i=0;i<arr.size();i++){
        cout<<arr[i]<<" ";
    }
    int c=0;
    mergesort(arr,c);
    cout<<endl;
    cout<<c;
}
```